process **901** can identify a "skeleton view" of the user based on traditional and/or depth data to include as a further or alternate data stream. A skeleton view divides the users into primary parts (e.g., hands, forearms, upper arms, shoulders, torso, etc.) and depicts these parts of the user as connected lines—giving a body pose of the user without body contours.

[0100] At block **906**, process **901** can apply tags to the captured data. In some implementations, these tags can be the calibration data generated by process **701**, where each data stream is tagged with the calibration data generated for the capture device that captured that data stream. In some implementations, other or additional tagging can be performed such as identifying objects or people within various portions of the captured data and tagging those portions with the identifications (e.g., by frame or by locations within frames), tagging conditions of the captured data (e.g., places where movement above a threshold is identified, changes in people in frame, etc.), identifying voices in audio and tagging time segments with the identified voices, etc.

[0101] FIG. **10** is a flow diagram illustrating a process **1001** (e.g., the process performed by block **508** of FIG. **5**) used in some implementations of the present technology for a compression stage of a 3D conversation pipeline. At block **1002**, process **1001** can receive the captured data, as filtered, enhanced, and/or tagged by process **901**.

[0102] At block **1004**, process **1001** can determine whether conversation context factors indicate whether to apply down sampling to the captured data. For example, the captured data can be down sampled to match the display capabilities of the receiving system or to reduce data complexity, thereby reducing the bandwidth needed to transmit the compressed data and reducing the compute power for the receiving device to decompress and reconstruct the data. In various examples, thresholds of available bandwidth, sender or receiver processing capabilities, or receiver display capabilities, can each be mapped to particular bitrates, frame rates, etc. At block **1004**, current conversation context factors can be identified and the mapping can be used to select maximum data characteristics. If the data received at block **1002** has higher characteristics than the selected maximum data characteristics then, at block **1006**, process **1001** can down sample the received data to the maximum data characteristics. In some implementations, the characteristics of the received data will already match capabilities of the receiving system because the capture devices were dynamically set to capture data with characteristics matching the receiving system.

[0103] While, as discussed above, in various implementations, any block from any of the flow charts can be removed or rearranged, blocks **1004** and **1006** are illustrated in broken lines to call out specific implementations where blocks **1004** and **1006** may or may not be performed. In some implementations, there is no dynamic down sampling and thus process **1001** can go from block **1002** to block **1008**.

[0104] At block **1008**, process **1001** can apply one or more compression procedures to the received data (as down sampled at block **1006**, if any). In some implementations, the down sampling of block **1006** can be performed as part of the compression performed at block **1008**. In various implementations, the compression procedures can be lossy or lossless. The compression procedures can be selected to match a type of the received data. For example, if part of the received data is a point cloud, then a compression procedure for point clouds (e.g., the MPEG codec for point clouds or the PCL) can be selected to compress this part of the received data. In various implementations, the received data from each capture device can be compressed separately and/or the body of all the captured data can be compressed into a single package. In some implementations, data from one part of the received data can be used to more effectively compress another part. For example, motion vectors from a data stream of a depth camera can be used to compress a color video data stream taken from a camera that is within a threshold position of the depth camera. The compressed data can then be transmitted to a receiving system or an intermediary server.

[0105] FIG. **11** is a flow diagram illustrating a process **1101** (e.g., the process performed by block **510** of FIG. **5**) used in some implementations of the present technology for a decompression stage of a 3D conversation pipeline. At block **1102**, process **1101** can receive data compressed by process **1001**, having been transmitted over a network.

[0106] At block **1104**, process **1101** can determine whether conversation context factors indicate whether to apply down sampling to the received data. For example, the received data can be down sampled to match the display capabilities or available compute power of the receiving device. In various examples, thresholds of receiver processing capabilities or display capabilities can each be mapped to particular bitrates, frame rates, etc. At block **1104**, current conversation context factors can be identified, and the mapping can be used to select these maximum data characteristics. If the data received at block **1102** has higher characteristics than the selected maximum data characteristics then, at block **1106**, process **1101** can down sample the received data to match the maximum data characteristics. In some implementations, the characteristics of the received data will already match capabilities of the receiving system because the capture devices were dynamically set to capture data with characteristics matching the receiving system or compression process **1001** already down sampled the data to match the receiving system.

[0107] While, as discussed above, in various implementations, any block from any of the flow charts can be removed or rearranged, blocks **1104** and **1106** are illustrated in broken lines to call out specific implementations where blocks **1104** and **1106** may or may not be performed. In some implementations, there is no dynamic down sampling and thus process **1101** can go from block **1102** to block **1108**.

[0108] At block **1108**, process **1101** can apply one or more decompression procedures to the received data (as down sampled at block **1106**, if any). In some implementations, the down sampling of block **1106** can be performed as part of the decompression performed at block **1108**. In various implementations, the decompression procedures can be lossy or lossless. The decompression procedures can be selected to match a type of compression used to compress the received data.

[0109] FIG. **12** is a flow diagram illustrating a process **1201** (e.g., the process performed by block **512** of FIG. **5**) used in some implementations of the present technology for a reconstruction stage of a 3D conversation pipeline. At block **1202**, process **1201** can receive data decompressed by process **1101**. At block **1204**, process **1201** can determine whether conversation context factors indicate whether to adjust the process for generating a 3D reconstruction based on available resources. In some implementations, types of